Cahier des charges

Projet tutoré - DUT Informatique

BENOIT Marine
DUROSIER Florian
NOWINSKI David
SID Ali
VOISIN Julien
ZAMBELLI Cédric

Table des matières

I. Pourquoi ce projet ?

Le but est de créer un logiciel éducatif de sensibilisation à la recherche opérationnelle. Ce logiciel permettra à des non-initiés (étudiants, entre autres) de s'initier de manière ludique aux problématiques de la recherche opérationnelle, qui font partie de notre vie de tous les jours.

II. Comment ?

Le projet s'articulera autour d'une application interactive. L'utilisateur aura à sa disposition trois sous-menus, chacun comportant plusieurs jeux basés sur des algorithmes de plus en plus difficiles. Ces jeux seront classés selon leur niveau de difficulté – d'un point de vue algorithmique :

- les algorithmes polynomiaux,
- les algorithmes pseudo-polynomiaux,
- les algorithmes exponentiels.

Le joueur essayera de trouver par lui-même les solutions aux différents problèmes proposés. La solution lui sera présentée de manière animée (par exemple, pour le plus court chemin, une recherche animée).

Les jeux continuent tant que le joueur n'a pas trouvé la solution idéale. Un bandeau, présent en haut de la fenêtre de jeu, initialement rouge avec un texte blanc « solution optimale non trouvée », passera au vert lors de la résolution (le texte changera en « vous avez trouvé, bravo ! »).

Le projet sera programmé avec le langage Python, afin d'assurer une portabilité vers tous les systèmes d'exploitation.

III. Description fonctionnelle

1) La page d'accueil

Elle doit comporter des liens ou boutons vers les trois catégories précédemment vues. Le design doit être attrayant et inviter les utilisateurs (qui sont, on le rappelle, supposés être novices et n'avoir que pas ou peu de connaissances en algorithmique et en recherche opérationnelle / optimisation) à s'intéresser au sujet.

2) La partie interactive

2.1) Temps polynomiaux

Le plus court chemin Le but du jeu du plus court chemin est d'aider un personnage à se rendre d'un point A à un point B en empruntant le chemin le plus court possible.

L'utilisateur pourra cliquer sur les différentes villes, tour à tour, pour déplacer le personnage. Un compteur kilométrique indiquera la distance parcourue.

Le joueur peut effectuer un clic droit pour annuler son dernier mouvement.

Le jeu continue tant que le joueur n'a pas trouvé le chemin le plus court. Il peut aussi cliquer sur « solution », qui affichera le chemin segment par segment.

Les niveaux sont générés aléatoirement.

Le couplage Le but du jeu est de partager un certain nombre d'objets avec plusieurs personnes. Chaque personne doit dresser une liste de ses objets préférés, et l'on doit distribuer de manière équitable les objets en essayant de satisfaire le plus grand nombre.

L'utilisateur pourra choisir différents niveaux ; le nombre d'objets changera en fonction du niveau de difficulté.

2.2) Temps pseudo-polynomiaux

Le problème du sac à dos Ce problème consiste à emporter des objets de valeurs différentes. Il s'agit d'optimiser la valeur des objets que l'on peut prendre en veillant à ne pas dépasser la capacité du sac à dos.

On pourra modéliser ce problème par un tableau contenant les objets et leur valeur en vis-à-vis. L'utilisateur fera glisser les objets du tableau vers le sac à dos. Un indicateur permettra de visualiser le poids actuel total des objets dans le sac, le poids maximal que peut supporter le sac, et le poids restant (différence entre ces deux résultats).

On pourra scénariser ce jeu avec, par exemple, une histoire où un personnage doit quitter son château assiégé en emportant un maximum d'objets de valeur.

Meilleur rendement dans une confiserie Dans ce jeu, l'utilisateur doit réussir à satisfaire au mieux les désirs d'une confiserie. La confiserie fabrique

trois types de bonbons : des caramels, des sucre d'orge et des chewing-gum. Chaque bonbon se vend à un prix qui lui est propre. On doit fabriquer une quantité minimale de chaque bonbon, mais on ne peut pas dépasser une certaine quantité maximale. On dispose également d'un certain temps imparti pour réaliser tous les bonbons ; chaque bonbon ayant évidemment un temps de fabrication différent.

La modélisation pourra s'effectuer à l'aide d'un tableau à double entrée présentant, en ligne, un bonbon et, en colonne, les différentes caractéristiques (temps de fabrication, nombres minimal et maximal). L'utilisateur devra cliquer sur les bonbons pour les fabriquer, une jauge de temps en haut de l'écran diminuera au fur et à mesure que le temps de fabrication des bonbons s'écoule. Là encore, l'utilisateur pourra revenir en arrière en effectuant un clic sur le bouton droit.

2.3) Temps exponentiels

Le problème du voyageur de commerce Un voyageur de commerce se situe dans une certaine ville A. Il doit passer par toutes les villes de la carte pour revenir à son point de départ. Le but est de déterminer le plus court chemin entre toutes les villes. On pourra reprendre le concept du premier jeu, en modifiant les distances entre les villes.

IV. Équipe

- Marine Benoit
- Florian Durosier
- David Nowinski
- Baptiste Valthier
- Julien Voisin
- Cédric Zambelli

Cette équipe risque d'être dissoute d'une part par d'éventuels redoublements de semestre 2, et d'autre part, par le départ de Baptiste Valthier à l'Université de Portsmouth dans le cadre d'un échange Erasmus.

V. Méthode de travail et organisation

Dans un premier temps, s'effectuera une phase de recherche d'informations et de documentation :

- documentation sur la recherche opérationnelle ;
- documentation sur les différents algorithmes utilisés ;
- recherches sur les librairies graphiques en Python ;
- approche de l'utilisation d'un logiciel de gestion de versions (tel que `mercurial`).

Le squelette de l'application fera l'objet d'une implémentation commune. Nous discuterons ensemble de la librairie graphique utilisée, des différentes conventions de codage et réaliserons ensemble le menu.

Les algorithmes seront répartis en groupes de deux personnes. Une fois le codage terminé, pourront s'ensuivre des échanges de code et des tests.

VI. Objectifs généraux

L'objectif est de créer une application *didactique*, qui s'articulera autour :

- d'un **menu général** ;
- de solutions **pas-à-pas** ;
- d'une génération **aléatoire** des jeux ;
- de différents **niveaux** de difficulté.

L'application devra être portable et multiplate-forme. Le langage de programmation retenu est le Python.

VII. Solutions alternatives

Si nous échouons à notre tâche, nous pourrons au choix supprimer certains algorithmes jugés trop difficiles à mettre en place, ou négliger l'aspect graphique afin de consacrer nos efforts au programme en lui-même.

VIII. Contrainte

Nous nous fixons la contrainte de terminer le projet en la période d'un semestre.